

آزمون میان نیم سال درس "طراحی و پیاده سازی زبان های برنامه سازی"

زمان آزمون: ۳ ساعت

تاریخ: ۱۴۰۰/۲/۳۰

نکات مهم:

- ۱- جواب تمام سوال ها را به صورت دست نوشته و یا تایپی بنویسید. سپس تمام جواب ها را به ترتیب شماره سوالات در یک فایل pdf با نام شماره دانشجوییتان گذاشته و در قسمت مربوطه در کوئرا آپلود کنید. دقت کنید که در کوئرا تنها امکان ارسال pdf وجود دارد. تنها ارسال نهایی شما در کوئرا تصحیح خواهد شد. دقت کنید که در صورت ارسال چند جواب، کوئرا به صورت خودکار آخرین ارسال را به عنوان ارسال نهایی در نظر می گیرد. برای رعایت عدالت و نظم، در زمان برگزاری آزمون به پرسشی پاسخ داده نخواهد شد.
- ۲- استفاده از هر منبع مکتوب یا اینترنتی در دسترس شخص دانشجو مجاز است. اما مشورت با دیگری (هر کسی که باشد و در هر اندازه) مجاز نیست. لطفا جواب سوال ها را به دیگران منتقل نکنید و از پاسخ دیگران حتی اگر در دسترس شما قرار گرفت استفاده یا کپی نکنید. در صورت وقوع چنین مواردی مطابق با آیین نامه های دانشگاه رفتار خواهد شد.

موفق، سلامت و پیروز باشید.

ایزدی

پرسش ها و مسائل:

۱. فرض کنید $isBigger?$ تابعی از پیش تعریف شده است که دو رشته ورودی را می گیرد و آنها را مقایسه می کند. به طور مثال در $(isBigger? \ s1 \ s2)$ در صورتی که $s1$ بزرگتر از $s2$ باشد، $\#t$ و در غیر این صورت $\#f$ برمی گرداند. برنامه ای به زبان رکت بنویسید که به عنوان ورودی یک لیست از رشته ها به همراه یک تابع $isBigger?$ دلخواه دریافت می کند و با کمک تابع $isBigger?$ لیست ورودی را به صورت صعودی مرتب کرده و مرتب شده ی آن را به عنوان خروجی برمی گرداند. در صورتی که دو رشته با هم برابر بودند، ترتیب قرار گیری آن ها در لیست خروجی اهمیتی ندارد.

نمونه ورودی:

`(main string>? ("a", "c", "b"))`

نمونه خروجی:

`'("a", "b", "c")`

۲. الف- برای مجموعه زیر به هر سه روش بالا به پایین، پایین به بالا و قوانین استنتاج تعریف های استقرایی را بنویسید.

$$\{(n, f(n), f(n+2)) \mid n \in \mathbb{N}, f(0) = 1, f(1) = 3, f(n+2) = f(n+1) + 2f(n)\}$$

ب- در هر مورد زیر بیان کنید که قوانین مشخص کننده ی چه مجموعه ای هستند.

- $(1, 2) \in S, \frac{(n, k) \in S}{(n+2, n+k) \in S}$
- $(0, 5) \in S, \frac{(n, k) \in S}{(n+2, 3k+1) \in S}$

۳. برنامه زیر به زبان PROC را در نظر بگیرید:

```
let fact = proc (n) -(n 1)
in let fact = proc (n)
    if zero?(n)
    then 0
    else *(n (fact -(n 1)))
in (fact 6)
```

الف- خروجی این برنامه چیست؟

ب- چرا دومین تابع fact در این برنامه، پیاده سازی درستی از تابع فاکتوریل در زبان PROC نیست؟ توضیح دهید.

ج- اگر بخواهیم بدون هیچ تغییری در نحو و دستور زبان PROC امکان نوشتن برنامه‌هایی بازگشتی مانند برنامه بالا با خروجی درست داشته باشیم، چه تغییری در معناشناسی یا توصیف ساختارهای زبان لازم است؟ در این صورت تغییرات لازم بر روی تمامی ساختارهای داده ای و تابعی که در نهایت به پیاده سازی تابع مفسر یعنی تابع value-of می‌شوند و همچنین تغییرات این تابع را بیان کنید.

۴. به زبان LET معرفی شده در کلاس، نوع داده ای رشته (string) را اضافه کنید. این نوع داده‌ی جدید فقط در عبارت let می‌تواند به یک متغیر نسبت داده شود. بقیه توابع روی این نوع داده عمل نمی‌کنند. دو عبارت (تابع) جدید *null?* و *in?* را برای این نوع داده به زبان اضافه و سپس آن را پیاده‌سازی کنید. *null?* در صورت خالی بودن رشته (رشته "") خروجی *#t* برمی‌گرداند و در غیر این صورت *#f* برمی‌گرداند. *in?* نیز دو رشته می‌گیرد و در صورتی که اولی درون دومی باشد (زیررشته‌ای از آن باشد) *#t* و در غیر این صورت *#f* برمی‌گرداند.

تمام تغییرات روی گرامر، توصیف مقادیر، محیط، قوانین استنتاج دستورات و کدهای تغییر یافته یا اضافه شده را مشخص نمایید. (بعضی از این موارد ممکن است تغییر نکنند).

۵. برنامه زیر را که به زبان PROC نوشته شده را در نظر بگیرید. خروجی این برنامه را مشخص کنید. تحلیل کنید اگر در خط آخر به جای ۵ عدد دیگری گذاشته می‌شد، خروجی چه بود.

```
let f = proc (maker)
    proc (x)
        if zero? (x)
        then 0
        else if zero? (-(x, 1))
        then 1
        else -(((maker maker) -(x, 1)), -(0, ((maker maker) -(x, 2))))
in let main = proc (x) ((f f) x)
in (main 5)
```

۶. به کمک روش سوال قبل و با استفاده از زبان PROC برنامه‌ای بنویسید که اول بودن یک عدد صحیح مثبت را بررسی کند و در صورت اول بودن *#t* و در غیر این صورت *#f* برگرداند.